**Lab 1 (Function Concepts)**

1. Write a program to add two numbers using functions.
2. Write a program to display sum of greatest number and smallest number among three given numbers using function.
3. Write a program to find xy using function
4. Write a program that calls variations of function called add(). This function can takes zero, one, two or three and prints the added value.
5. Write a program that calls a function called getarea (). This function is responsible for reading two integer numbers (length and breadth) and prints its area.
6. A point has x-axis and y-axis. Define a structure called ‘point’ with suitable members. Write a program that accepts two points form the user and calculates the distance between these two points.

**Lab 2 (Function Overloading and Inline Functions)**

1. Write a function called power() that raises a number n to a power m where both m and n are integers. Use a default argument of 2 for m. The function power should return the result of the calling function.
2. Write a function called multiply() that takes in two arguments and then print the result. Overload the function to a) that takes no arguments b) that takes three arguments.
3. Write a function to find the area of square, rectangle and circle showing function overloading.
4. Use the concept of function overloading to calculate
   1. Volume of cylinder
   2. Volume of cuboid
   3. Volume of cube
5. Write a function called multiply() that takes in two arguments and then returns the multiplied value of two variables. Make the function inline.
6. Write a function that finds the largest of three numbers. Make the function inline.
7. Using inline function convert Fahrenheit temperature to centigrade.
8. Using inline function covert pounds to kilogram (Hint: 1 kg=0.453592 pounds)

## **Lab 3 (Class and Objects)**

1. Write a simple class **circle** with one membervariable**,** radius. Include a member function getradius() to get data from the user and another function printarea() to display the area.
2. Write a simple class **square** with one member variable, length. Include a member function getlength() to get the length from the user and two another functions printarea() and printperimeter() to display the area and perimeter of the square respectively.
3. Write a C++ program to represent a class called **large.** Include two data members to store numbers. Use member functions **get()**,**largest()** and **put()** to get data, to find largest of two and finally to display largest.

Create an instance of above class to perform these tasks.

1. Create a class to represent batsman in a cricket team which includes the following members:

**Data member:**

Name, Runs made, No-of-fours, No-of-sixes

**Member function**:

To assign the initial value

To display batsman information

Create an array of object to display the information of five batsmen.

1. Create a class called **employee** which contains private data and member function like: Employee-Name, Employee- address, Employee-Phone, Salary, Net-salary, tax and **calculate(),** to calculate the Net-salary i.e.(Net-salary=Salary-tax) and public member functions like **input()** and **output()** to get the information of employee and display the information along with the calculated Net-salary value respectively.

Now, write a main() program to create an array of object to handle ‘n’ employee.

1. Write a class **item** with data item **number** and a static member variable **count**. Define a member function **getdata()** to accept input from user as well as count the number of times this function is called by different objects. Write a main function to exercise this class.
2. Create two classes **ABC** and **XYZ**, having one integer member variable and a member function to get the input. Now, create a function **max()** which takes as arguments objects of **ABC** and **XYZ** and displays the greater of the two. Declare this function friend functions in both classes.
3. Create a class called **time** with private data items hours, minutes and seconds. Write inside the class an input function that accepts inputs from the user. Write another outside the class definition that takes as arguments two objects of class **time** and returns another **time** object that holds the sum of the two **time** variables passed as arguments.
4. Imagine a tollbooth at a bridge .Cars passing by the booth is expected to pay five rupees. Mostly they do, but sometimes a car goes by without paying. The tollbooth keeps track of the number of cars that have gone by, and of the total amount of money collected.

Model this tollbooth with a class called **tollbooth**. The two data items are type **int** to hold the total number of cars, and a type **double** to hold the total amount of money collected. A member function called **initialize()** initialize both these to zero. A member function called **payingcar()** increments the car total and adds Rs five to the cash total. Another function, called **nopaycar()**, increments the car total but adds nothing to the cash total. Finally, a member function called **display()** displays the two totals.

Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the **ESC** key should cause the program to print out the total cars and total cash and then exit.

1. Calculate the mean of two private values of class **sample** using friend function.
2. Write a C++ program to swap the private values of two classes using friend function.
3. Write a C++ program to add time in the hour and minutes format. Use Objects as function arguments to perform the task.
4. Write a C++ program to add two complex numbers using friend function. Use the concept of passing of arguments to the function and returning objects to perform the task.

# **Lab 4 (Constructors)**

1. Write a program to store your name and salary in an Employee object. (Don’t prompt for input; just initialize the object in the program). Output your name and current salary. Then give yourself a 2,500 pay rise and output your new salary.

1. Write a program that creates two simple circle objects. Use default constructor on one and initiate the other with the value 7. Then print the area of each circle.
2. Write a program that copies data member of one object to another with the help of a copy constructor.
3. Create a class Int (note different spelling). The only data in this class is an int variable. Include member functions to initialize an Int to 0, to initialize it to an int value, to display it, and to add two Int values.

Write a program that exercises this class by creating two initialized values an placing the response in the initialized value, and then displaying the result.

1. Create a class called time that has separate int member data for hours, minutes, and seconds. One constructor should initialize this data to 0 and another should initialize it to fixed values. A member function should display it in hh:mm:ss format. The final member function should add two objects of time passed as arguments. Create two, initialized time objects in the main program and one that isn’t initialized. Then it should add the two initialized values together leaving the result in the third time variable.
2. Write a constructor CHECK that initializes an integer variables x,y and prints “This is COSTRUCTOR example”. Print the value of x and y using display function.
3. Write a constructor INI that initializes a variable to 15 and destructor that destroy it.
4. Write a constructor function test that initializes integer variable a to 10 and b to 20 and print “This is constructor example” and member functions,

Inc that increment the variable a and b by 5.

Dec that decrements the variable a and b by 3.

Ret that returns the sum of a and b.

**Lab 5 (Operators Overloading)**

1. Create a class **item** with a single integer member data count. Overload the ++operator allowing operations like c1++, c2=c1++ where c1,c2 are objects of the class **item**.
2. Create a class **item** with a single integer member data value. Overload the unary minus (-) operator allowing operations like P=-Q where P, Q are objects of the class **item**.
3. Write a program that overloads the + operator to concatenate two strings such as str3=str1+str1
4. Write a program to overload “>=” operator.
5. Define a class string. Overload the = = operator to compare two strings.
6. Define a class **time** and instead of using a function to add time, use the overloaded operator to add two times.
7. Overload the binary – operator which will work for the operation like A=B-C, where A, B, C are the object of the same class.
8. Write a program to overload \*= operator which should allow statements like L1\*=L2;

Where L1,L2 are the object of a class length which has a single int variable as its data member.

1. Write a program to convert Miles class into Meter class and vice versa.
2. Write a program to Display the area of Rect object, using cout<<Object\_name.
3. Write a program to Create a class square, make few instances, and assign the integers value to each of them directly.

**Lab 6 (Inherintence)**

1. Create a class **students** with data members **name** and **roll**, and member functions **getdata()**, **display()**. From **student**, derive a class **result** with data member that stores the marks of two subjects, and member functions **getmarks()** to enter marks, **marksheet()** to display the result. Write a suitable main program to read and display name, roll and sum of marks1 and marks2 of 5 students.

1. Imagine a publishing company that markets both book and audio cassette versions of its works. Create a class **publication** that stores a **title** (a string) and **price** (type float) of a publication. From this class derive two classes: class **book,** which adds a page count (type int); and class **tape**, which adds a playing time minutes(type float). Each of these three classes should have a **getdata()** function to gets data from the user at the keyboard, and the **putdata()** function to display its data.



1. Create a class student that stores name ( a string) and roll no (type int). From this class derive a class marks that adds sub1 (type float), sub2 (type float) that stores the marks for two subjects. Then form the class marks derive a class record which adds semester type (type int) and average (type float), this average is the average marks of the sub1 and sub2 defined in class marks.



1. An educational institution wishes to maintain a database of its employees. The database is divided into a number of classes whose hierarchical relationships are shown in the figure. The figure also shows the minimum information required for each class. Specify all the classes and define functions to create the database and retrieve individual information as and when required.













1. Consider a class network of the figure. The class master derives information from both account and admin classes which in turn derive information from the class person. Define all the four classes and write a program to create, update and display the information contained in master objects.













1. Modify question 1 to include constructors for all the three classes.

### **Lab 7 (File Handling)**

1. Write a program to write, “I have to work hard to be success” to a file and read from the file using a constructor.
2. Write a program to write a character file to a file and read the character from a file.
3. Write a program to
   1. Enter the name and telephone numbers of peoples
   2. Stores it in file.
   3. Display it in following format.

| Name | Number |
| --- | --- |
| Ram | 234567 |
| Hari | 123456 |
| ……… | ………… |
| ……….. | ………… |

* 1. Search the name, when the number is given.

1. Write a program that accepts the player’s name and score and stores the result into a file. Also display all the data of file.
2. A file contains a list item name, item code and cost in the following form:

| Name | Code | Cost |
| --- | --- | --- |
| Mango | 1001 | 100.00 |
| Orange | 1002 | 150.00 |
| Apple | 1003 | 200.00 |

Write a program to read the file and output the list in three columns. The name should be left justified and cost right justified.

1. Write a program to create a file “student.txt” in which the following operations are performed:
   1. Add a new student in the file.
   2. Modify the details of the students.
   3. Display the content of file.

LAB-8 (Templates and Namespaces)

1. Define a template function swapValues that swaps two values of any type. Test it with integers, floating-point numbers, and strings.
2. Implement a template class Pair that holds a pair of values of any type. Provide methods to get and set the values in the pair. Test the class with different data types.
3. Write a template function findMax that returns the maximum of two values of any type. Overload this function to handle finding the maximum of three values.
4. Demonstrate the usage of both versions of findMax with different data types.

Dear all(BE Comuter-II ),,

Please find the link below. And fill up the form before sunday.

Use your KCC credentials to log in

<https://forms.gle/hUWTGdB1b5KXva5j6>